

CLAIMS

1. A method for producing embedded software, comprising:

providing one or more generic application handler programs, each such program comprising computer program code for performing generic application functions common to multiple types of hardware modules used in a communication system;

generating specific application handler code to associate the generic application functions with specific functions of a device driver for at least one of the types of the hardware modules; and

compiling the generic application handler programs together with the specific application handler code to produce machine-readable code to be executed by an embedded processor in the at least one of the types of the hardware modules.

2. A method according to claim 1, wherein providing the generic application handler programs comprises providing an application program interface (API) to enable a system management program in the communication system to invoke the generic application functions.

3. A method according to claim 2, wherein the one or more generic application handler programs comprise a plurality of generic application programs, and wherein providing the API comprises enabling one of the generic application programs to invoke the generic application functions of another of the generic application programs.

4. A method according to claim 1, wherein providing the generic application handler programs comprises providing a performance monitoring handler, including a performance

monitoring function for counting selected events relating to performance of the hardware modules.

5. A method according to claim 4, wherein generating the specific application handler code comprises specifying a register in one of the types of the hardware modules whose contents are to be passed to the performance monitoring function for counting.

6. A method according to claim 4, wherein providing the generic application handler programs further comprises providing an alarm handler, and wherein providing the performance monitoring handler comprises providing a programmable performance threshold and an alarm invocation function, such that when a count of the selected events exceeds the threshold, the performance monitoring handler sends an alarm message to the alarm handler.

7. A method according to claim 1, wherein providing the generic application handler programs comprises providing a maintenance handler, including a testing function for detecting failures in the hardware modules.

8. A method according to claim 7, wherein generating the specific application handler code comprises associating the testing function with at least one of a self test and a sanity test of a component in the at least one of the types of the hardware modules.

9. A method according to claim 7, wherein providing the generic application handler programs further comprises providing an alarm handler, and wherein providing the maintenance handler comprises providing an alarm invocation function, such that when one of the failures

is detected, the maintenance handler sends an alarm message to the alarm handler.

10. A method according to claim 1, wherein providing the generic application handler programs comprises providing a configuration handler, for holding configuration and state information regarding components of the hardware modules.

11. A method according to claim 1, wherein providing the generic application handler programs comprises providing an alarm handler, including functions for receiving and responding to alarm messages generated by others of the application handler programs.

12. A method according to claim 11, wherein providing the alarm handler comprises providing a programmable prioritization function, for determining an order of priority for processing of the alarm messages by the alarm handler.

13. A method according to claim 11, wherein generating the specific application handler code comprises specifying a component in one of the types of the hardware modules to actuate so as to notify a user of the system of the alarm message.

14. A method according to claim 11, wherein generating the specific application handler code further comprises specifying one of the generic application functions of another of the generic application handler programs to invoke in response to the alarm message.

15. A method according to claim 11, wherein responding to the alarm messages comprises sending a notification of

the alarm message from the alarm handler to a system management program.

16. A method according to claim 1, wherein generating the specific application handler code comprises defining specific elements to be handled by the generic application functions for the at least one of the types of the hardware modules, and registering one of the specific functions of the device driver for use in handing each of the defined specific elements.

17. A hardware module for use in a communication system, belonging to a particular type of module among multiple types of hardware modules used in the communication system, the module comprising:

communication circuitry, comprising hardware components for processing communication signals;

an embedded central processing unit (CPU), coupled to control the components of the communication circuitry and to receive data therefrom; and

a memory, containing machine-readable code for execution by the CPU, the machine-readable code having been generated by compiling together one or more generic application handler programs, each such program comprising computer program code for performing generic application functions common to the multiple types of hardware modules used in the communication system, along with specific application handler code prepared so as to associate the generic application functions with specific functions of a device driver for the particular type of module to which the hardware module belongs.

18. A module according to claim 17, wherein the generic application handler programs comprise an application

program interface (API) to enable a system management program in the communication system to invoke the generic application functions.

19. A module according to claim 18, wherein the one or more generic application handler programs comprise a plurality of generic application programs, and wherein the API enables one of the generic application programs to invoke the generic application functions of another of the generic application programs.

20. A module according to claim 17, wherein the generic application handler programs comprise a performance monitoring handler, including a performance monitoring function for counting selected events relating to performance of the hardware modules.

21. A module according to claim 20, wherein the specific application handler code specifies a register in one of the types of the hardware modules whose contents are to be passed to the performance monitoring function for counting.

22. A module according to claim 20, wherein the generic application handler programs further comprise an alarm handler, and wherein the performance monitoring handler provides a programmable performance threshold and an alarm invocation function, such that when a count of the selected events exceeds the threshold, the performance monitoring handler sends an alarm message to the alarm handler.

23. A module according to claim 17, wherein the generic application handler programs comprise a maintenance

handler, including a testing function for detecting failures in the hardware modules.

24. A module according to claim 23, wherein the specific application handler code associates the testing function with at least one of a self test and a sanity test of one of the hardware components in the hardware module.

25. A module according to claim 23, wherein the generic application handler programs further comprise an alarm handler, and wherein the maintenance handler provides an alarm invocation function, such that when one of the failures is detected, the maintenance handler sends an alarm message to the alarm handler.

26. A module according to claim 17, wherein the generic application handler programs comprise a configuration handler, for holding configuration and state information regarding the hardware components.

27. A module according to claim 17, wherein the generic application handler programs comprise an alarm handler, including functions for receiving and responding to alarm messages generated by others of the application handler programs.

28. A module according to claim 27, wherein the alarm handler comprises a programmable prioritization function, for determining an order of priority for processing of the alarm messages by the alarm handler.

29. A module according to claim 27, wherein the specific application handler code specifies a component in one of the types of the hardware modules to actuate so as to notify a user of the system of the alarm message.

30. A module according to claim 27, wherein the specific application handler code further specifies one of the generic application functions of another of the generic application handler programs to invoke in response to the alarm message.

31. A module according to claim 27, wherein the alarm handler causes the CPU to send a notification of the alarm message to a system management program.

32. A module according to claim 17, wherein the specific application handler code defines specific elements to be handled by the generic application functions for the particular type of the hardware module, and registers one of the specific functions of the device driver for use in handing each of the defined specific elements.

33. A computer software product, comprising a computer-readable medium in which program code is stored, the code comprising one or more generic application handler programs, each such application handler program comprising program instructions for performing generic application functions common to multiple types of hardware modules used in a communication system, the generic application handler programs being arranged to be compiled together with specific application handler code prepared so as to associate the generic application functions with specific functions of a device driver for at least one of the types of the modules, thereby to produce machine-readable code to be executed by an embedded processor in the at least one of the types of the hardware modules.

34. A product according to claim 33, wherein the generic application handler programs comprise an application

program interface (API) to enable a system management program in the communication system to invoke the generic application functions.

35. A product according to claim 34, wherein the one or more generic application handler programs comprise a plurality of generic application programs, and wherein the API enables one of the generic application programs to invoke the generic application functions of another of the generic application programs.

36. A product according to claim 33, wherein the generic application handler programs comprise a performance monitoring handler, including a performance monitoring function for counting selected events relating to performance of the hardware modules.

37. A product according to claim 36, wherein the specific application handler code specifies a register in one of the types of the hardware modules whose contents are to be passed to the performance monitoring function for counting.

38. A product according to claim 36, wherein the generic application handler programs further comprise an alarm handler, and wherein the performance monitoring handler provides a programmable performance threshold and an alarm invocation function, such that when a count of the selected events exceeds the threshold, the performance monitoring handler sends an alarm message to the alarm handler.

39. A product according to claim 33, wherein the generic application handler programs comprise a maintenance

handler, including a testing function for detecting failures in the hardware modules.

40. A product according to claim 39, wherein the specific application handler code associates the testing function with at least one of a self test and a sanity test of one of the hardware components in the hardware module.

41. A product according to claim 39, wherein the generic application handler programs further comprise an alarm handler, and wherein the maintenance handler provides an alarm invocation function, such that when one of the failures is detected, the maintenance handler sends an alarm message to the alarm handler.

42. A product according to claim 33, wherein the generic application handler programs comprise a configuration handler, for holding configuration and state information regarding the hardware components.

43. A product according to claim 33, wherein the generic application handler programs comprise an alarm handler, including functions for receiving and responding to alarm messages generated by others of the application handler programs.

44. A product according to claim 43, wherein the alarm handler comprises a programmable prioritization function, for determining an order of priority for processing of the alarm messages by the alarm handler.

45. A product according to claim 43, wherein the specific application handler code specifies a component in one of the types of the hardware modules to actuate so as to notify a user of the system of the alarm message.

46. A product according to claim 43, wherein the specific application handler code further specifies one of the generic application functions of another of the generic application handler programs to invoke in response to the alarm message.

47. A product according to claim 43, wherein when the code is executed by the embedded processor, the alarm handler causes the processor to send a notification of the alarm message to a system management program.

48. A product according to claim 33, wherein the specific application handler code defines specific elements to be handled by the generic application functions for the particular type of the hardware module, and registers one of the specific functions of the device driver for use in handling each of the defined specific elements.